REJ10J1273-0200

Everywhere you imagine.

SuperH™ Family E10A-USB Emulator

Additional Document for User's Manual

Supplementary Information on Using the SH7211

Renesas Microcomputer Development Environment System

SuperH™ Family / SH7210 Series

E10A-USB for SH7211 HS7211KCU01HE

User's Manual

Rev.2.00
Revision Date: Apr. 10, 2007

Renesas Technology
www.renesas.com

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (http://www.renesas.com)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
   (1) artificial life support devices or systems
   (2) surgical implantations
   (3) healthcare intervention (e.g., excision, administration of medication, etc.)
   (4) any other purposes that pose a direct threat to human life

   Renesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.

Contents

Section 1 Connecting the Emulator with the User System 1
1.1 Components of the Emulator 1
1.2 Connecting the Emulator with the User System 2
1.3 Installing the H-UDI Port Connector on the User System 3
1.4 Pin Assignments of the H-UDI Port Connector 4
1.5 Recommended Circuit between the H-UDI Port Connector and the MCU 8
1.5.1 Recommended Circuit (36-Pin Type) 8
1.5.2 Recommended Circuit (14-Pin Type) 10
1.5.3 Recommended Circuit (38-Pin Type) 12

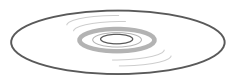
Section 2 Software Specifications when Using the SH7211 15
2.1 Differences between the MCU and the Emulator 15
2.2 Specific Functions for the Emulator when Using the SH7211 21
2.2.1 Event Condition Functions 21
2.2.2 Trace Functions 27
2.2.3 Note on Using the JTAG (H-UDI) Clock (TCK) 39
2.2.4 Notes on Setting the [Breakpoint] Dialog Box 39
2.2.5 Notes on Setting the [Event Condition] Dialog Box and the BREAKCONDITION_ SET Command 40
2.2.6 Performance Measurement Function 40

Section 1 Connecting the Emulator with the User System

1.1 Components of the Emulator

The E10A-USB emulator supports the SH7211. Table 1.1 lists the components of the emulator.

Table 1.1 Components of the Emulator

Classification	Component	Appearance	Quantity	Remarks
Hardware	Emulator box		1	HS0005KCU01H: Depth: 65.0 mm, Width: 97.0 mm, Height: 20.0 mm, Mass: 72.9 g or HS0005KCU02H: Depth: 65.0 mm, Width: 97.0 mm, Height: 20.0 mm, Mass: 73.7 g
	User system interface cable		1	14-pin type: Length: 20 cm, Mass: 33.1 g
	User system interface cable		1	36-pin type: Length: 20 cm, Mass: 49.2 g (only for HS0005KCU02H)
	USB cable		1	Length: 150 cm, Mass: 50.6 g
Software	E10A-USB emulator setup program, SuperH™ Family E10A-USB Emulator User's Manual, Supplementary Information on Using the SH7211*, and Test program manual for HS0005KCU01H and HS0005KCU02H		1	HS0005KCU01SR, HS0005KCU01HJ, HS0005KCU01HE, HS7211KCU01HJ, HS7211KCU01HE, HS0005TM01HJ, and HS0005TM01HE (provided on a CD-R)

Note: Additional document for the MCUs supported by the emulator is included. Check the target MCU and refer to its additional document.

1.2 Connecting the Emulator with the User System

To connect the E10A-USB emulator (hereinafter referred to as the emulator), the H-UDI port connector must be installed on the user system to connect the user system interface cable. When designing the user system, refer to the recommended circuit between the H-UDI port connector and the MCU. In addition, read the E10A-USB emulator user's manual and hardware manual for the related device.

Table 1.2 shows the type number of the emulator, the corresponding connector type, and the use of AUD function.

Table 1.2 Type Number, AUD Function, and Connector Type

Type Number	Connector	AUD Function
HS0005KCU02H	36-pin connector	Available
HS0005KCU01H, HS0005KCU02H	14-pin connector	Not available
HS0005KCU02H	38-pin connector	Available

The H-UDI port connector has the 36-pin, 14-pin, and 38-pin types as described below. Use them according to the purpose of the usage.

1. 36-pin type (with AUD function)
   The AUD trace function is supported. A large amount of trace information can be acquired in realtime. The window trace function is also supported for acquiring memory access in the specified range (memory access address or memory access data) by tracing.
2. 14-pin type (without AUD function)
   The AUD trace function cannot be used because only the H-UDI function is supported. Since the 14-pin type connector is smaller than the 36-pin type (1/2.5), the size of the area where the connector is installed on the user system can be reduced.
3. 38-pin type (with AUD function)
   The AUD trace function is supported. As well as the 36-pin type, a large amount of trace information can be acquired in realtime. Since the 38-pin type connector is smaller than the 36-pin type (1/2.5), the size of the area where the connector is installed on the user system can be reduced. To use the 38-pin type connector, however, an optional cable (HS0005ECK01H) is required.

1.3 Installing the H-UDI Port Connector on the User System

Table 1.3 shows the recommended H-UDI port connectors for the emulator.

Table 1.3 Recommended H-UDI Port Connectors

Connector	Type Number	Manufacturer	Specifications
36-pin connector	DX10M-36S	Hirose Electric Co., Ltd.	Screw type
	DX10M-36SE, DX10G1M-36SE		Lock-pin type
14-pin connector	2514-6002	Minnesota Mining & Manufacturing Ltd.	14-pin straight type
38-pin connector	2-5767004-2	Tyco Electronics AMP K.K.	38-pin Mictor type

Note: When designing the 36-pin connector layout on the user board, do not connect any components under the H-UDI connector. When designing the 14-pin connector layout on the user board, do not place any components within 3 mm of the H-UDI port connector. When designing the 38-pin connector layout on the user board, reduce cross-talk noise etc. by keeping other signal lines out of the region where the H-UDI port connector is situated. As shown in figure 1.1, an upper limit (5 mm) applies to the heights of components mounted around the user system connector.

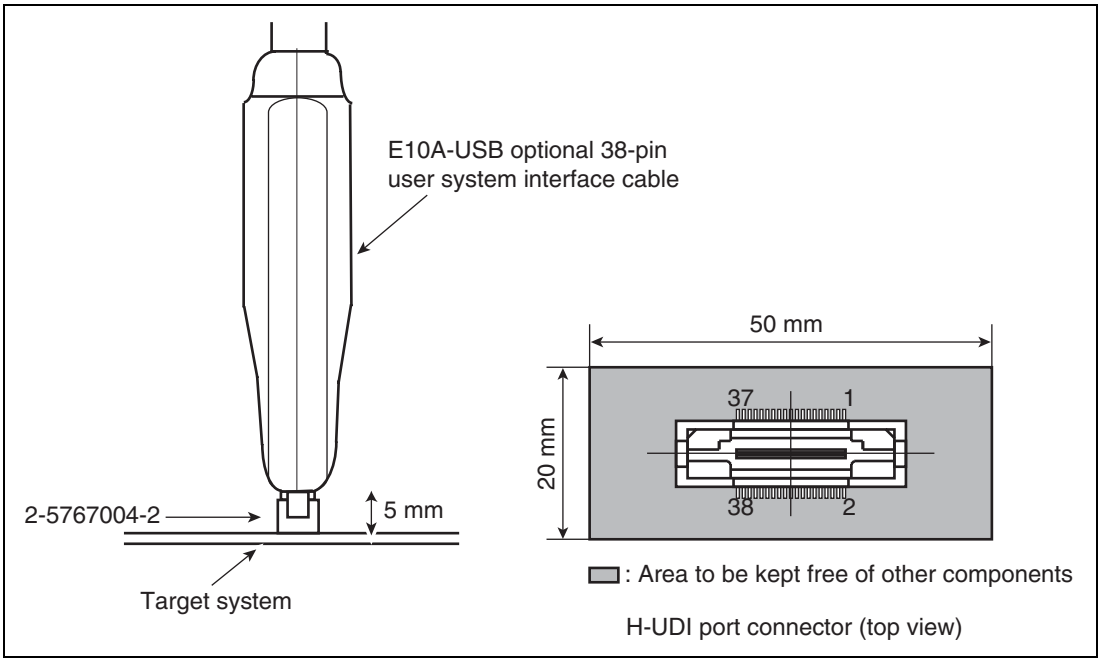


Figure 1.1 Restriction on Component Mounting

1.4 Pin Assignments of the H-UDI Port Connector

Figures 1.2 through 1.4 show the pin assignments of the 36-pin, 14-pin, and 38-pin H-UDI port connectors, respectively.

Note: Note that the pin number assignments of the H-UDI port connector shown on the following pages differ from those of the connector manufacturer.

Pin No.	Signal	Input/ Output*1	SH7211 Pin No.	Note	Pin No.	Signal	Input/ Output*1	SH7211 Pin No.	Note
1	AUDCK	Output	110		19	TMS	Input	17	
2	GND	——			20	GND	——		
3	AUDATA0	Output	116		21	_TRST*2	Input	18	
4	GND	——			22	(GND)*4	——		
5	AUDATA1	Output	115		23	TDI	Input	12	
6	GND	——			24	GND	——		
7	AUDATA2	Output	112		25	TDO	Output	15	
8	GND	——			26	GND	——		
9	AUDATA3	Output	111		27	_ASEBRKAK /_ASEBRK*2	Input/ output	30	
10	GND	——			28	GND	——		
11	_AUDSYNC*2	Output	109		29	UVCC	Output		
12	GND	——			30	GND	——		
13	N.C.	——			31	_RES*2	Output	22	User reset
14	GND	——			32	GND	——		
15	N.C.	——			33	GND*3	Output		
16	GND	——			34	GND	——		
17	TCK	Input	16		35	N.C.	——		
18	GND	——			36	GND	——		

Notes: 1. Input to or output from the user system.
2. The symbol (_) means that the signal is active-low.
3. The emulator monitors the GND signal of the user system and detects whether or not the user system is connected.
4. When the user system interface cable is connected to this pin and the _ASEMD pin is set to 0, do not connect to GND but to the _ASEMD pin directly.

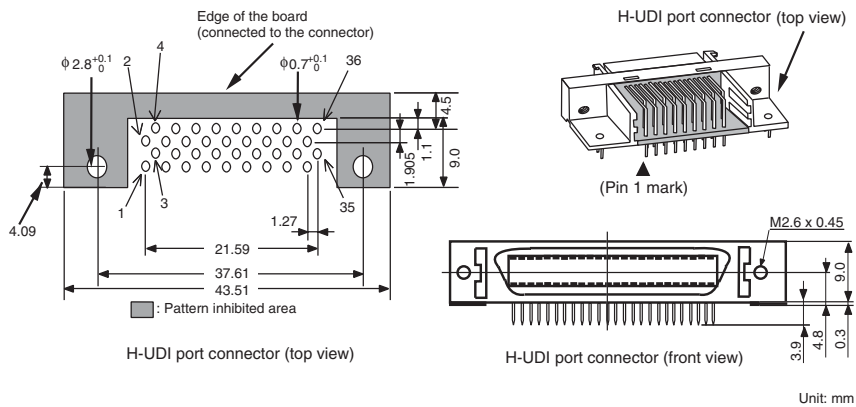


Figure 1.2 Pin Assignments of the H-UDI Port Connector (36 Pins)

Pin No.	Signal		Input/ Output*1	SH7211 Pin No.	Note
1	TCK		Input	16	
2	_TRST	*2	Input	18	
3	TDO		Output	15	
4	_ASEBRKAK / _ASEBRK	*2	Input/ output	30	
5	TMS		Input	17	
6	TDI		Input	12	
7	_RES	*2	Output	22	User reset
8	N.C.		—		
9	(GND)	*4	—		
11	UVCC		Output		
10, 12, and 13	GND		—		
14	GND	*3	Output		

Notes: 1. Input to or output from the user system.
2. The symbol (_) means that the signal is active-low.
3. The emulator monitors the GND signal of the user system and detects whether or not the user system is connected.
4. When the user system interface cable is connected to this pin and the _ASEMD pin is set to 0, do not connect to GND but to the _ASEMD pin directly.

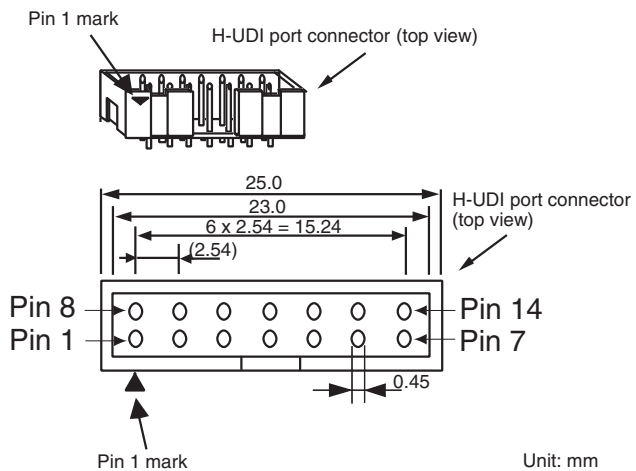
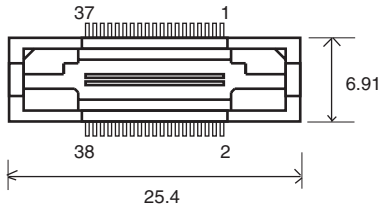


Figure 1.3 Pin Assignments of the H-UDI Port Connector (14 Pins)

Pin No.	Signal	Input/ Output*1	SH7211 Pin No.	Note	Pin No.	Signal	Input/ Output*1	SH7211 Pin No.	Note
1	N.C.	—			20	N.C.	—		
2	N.C.	—			21	_TRST*2	Input	18	
3	_ASEMD (GND)*4	—			22	N.C.	—		
4	N.C.	—			23	N.C.	—		
5	_UCON (GND) *3	—			24	AUDATA3	Output	111	
6	AUDCK	Output	110		25	N.C.	—		
7	N.C.	—			26	AUDATA2	Output	112	
8	_ASEBRKAK/ _ASEBRK*2	Input/ Output	30		27	N.C.	—		
9	_RES*2	Output	22	User reset	28	AUDATA1	Output	115	
10	N.C.	—			29	N.C.	—		
11	TDO	Output	15		30	AUDATA0	Output	116	
12	UVCC_AUD	Output			31	N.C.	—		
13	N.C.	—			32	_AUDSYNC	Output	109	
14	UVCC	Output			33	N.C.	—		
15	TCK	Input	16		34	N.C.	—		
16	N.C.	—			35	N.C.	—		
17	TMS	Input	17		36	N.C.	—		
18	N.C.	—			37	N.C.	—		
19	TDI	Input	12		38	N.C.	—		

Notes: 1. Input to or output from the user system.
2. The symbol (_) means that the signal is active-low.
3. The emulator monitors the GND signal of the user system and detects whether or not the user system is connected.
4. When the user system interface cable is connected to this pin and the _ASEMD pin is set to 0, do not connect to GND but to the _ASEMD pin directly.
5. The GND bus lead at the center of the H-UDI port connector must be grounded.



H-UDI port connector (top view)

Figure 1.4 Pin Assignments of the H-UDI Port Connector (38 Pins)

1.5 Recommended Circuit between the H-UDI Port Connector and the MCU

1.5.1 Recommended Circuit (36-Pin Type)

Figure 1.5 shows a recommended circuit for connection between the H-UDI and AUD port connectors (36 pins) and the MCU when the emulator is in use.

Notes: 1. Do not connect anything to the N.C. pins of the H-UDI port connector.

2. The _ASEMD pin must be 0 when the emulator is connected and 1 when the emulator is not connected, respectively.
   (1) When the emulator is used: _ASEMD = 0
   (2) When the emulator is not used: _ASEMD = 1
   Figure 1.5 shows an example of circuits that allow the _ASEMD pin to be GND (0) whenever the emulator is connected by using the user system interface cable.
   When the _ASEMD pin is changed by switches, etc., ground pin 22. Do not connect this pin to the _ASEMD pin.
3. When a network resistance is used for pull-up, it may be affected by a noise. Separate TCK from other resistances.
4. The pattern between the H-UDI port connector and the MCU must be as short as possible. Do not connect the signal lines to other components on the board.
5. The AUD signals (AUDCK, AUDATA3 to AUDATA0, and _AUDSYNC) operate in high speed. Isometric connection is needed if possible. Do not separate connection nor connect other signal lines adjacently.
6. Since the H-UDI and the AUD of the MCU operate with the VccQ, supply only the VccQ to the UVCC pin. Make the emulator's switch settings so that the user power will be supplied (SW2 = 1 and SW3 = 1).
7. The resistance value shown in figure 1.5 is for reference.
8. For the AUDCK pin, guard the pattern between the H-UDI port connector and the MCU at GND level.
9. The _TRST pin must be at the low level for a certain period when the power is supplied whether the H-UDI is used or not.
10. For the pin processing in cases where the emulator is not used, refer to the hardware manual of the related MCU.

When the circuit is connected as shown in figure 1.5, the switches of the emulator are set as SW2 = 1 and SW3 = 1. For details, refer to section 3.8, Setting the DIP Switches, in the SuperH™ Family E10A-USB Emulator User's Manual.

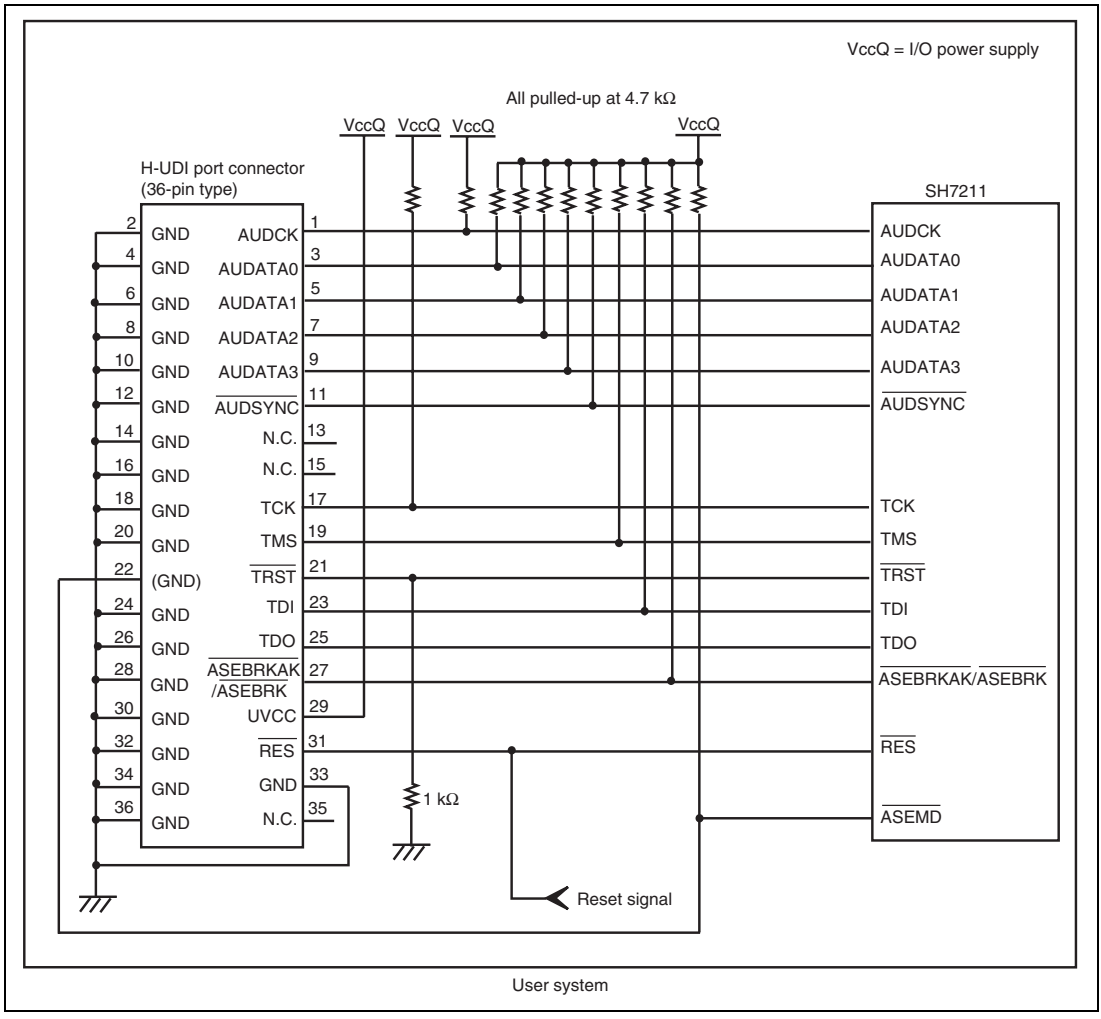


Figure 1.5 Recommended Circuit for Connection between the H-UDI Port Connector and MCU when the Emulator is in Use (36-Pin Type)

1.5.2 Recommended Circuit (14-Pin Type)

Figure 1.6 shows a recommended circuit for connection between the H-UDI port connector (14 pins) and the MCU when the emulator is in use.

Notes: 1. Do not connect anything to the N.C. pins of the H-UDI port connector.

2. The _ASEMD pin must be 0 when the emulator is connected and 1 when the emulator is not connected, respectively.
   (1) When the emulator is used: _ASEMD = 0
   (2) When the emulator is not used: _ASEMD = 1
   Figure 1.6 shows an example of circuits that allow the _ASEMD pin to be GND (0) whenever the emulator is connected by using the user system interface cable.
   When the _ASEMD pin is changed by switches, etc., ground pin 9. Do not connect this pin to the _ASEMD pin.
3. When a network resistance is used for pull-up, it may be affected by a noise. Separate TCK from other resistances.
4. The pattern between the H-UDI port connector and the MCU must be as short as possible. Do not connect the signal lines to other components on the board.
5. Since the H-UDI of the MCU operates with the VccQ, supply only the VccQ to the UVCC pin. Make the emulator's switch settings so that the user power will be supplied (SW2 = 1 and SW3 = 1).
6. The resistance value shown in figure 1.6 is for reference.
7. The _TRST pin must be at the low level for a certain period when the power is supplied whether the H-UDI is used or not.
8. For the pin processing in cases where the emulator is not used, refer to the hardware manual of the related MCU.

When the circuit is connected as shown in figure 1.6, the switches of the emulator are set as SW2 = 1 and SW3 = 1. For details, refer to section 3.8, Setting the DIP Switches, in the SuperH™ Family E10A-USB Emulator User's Manual.

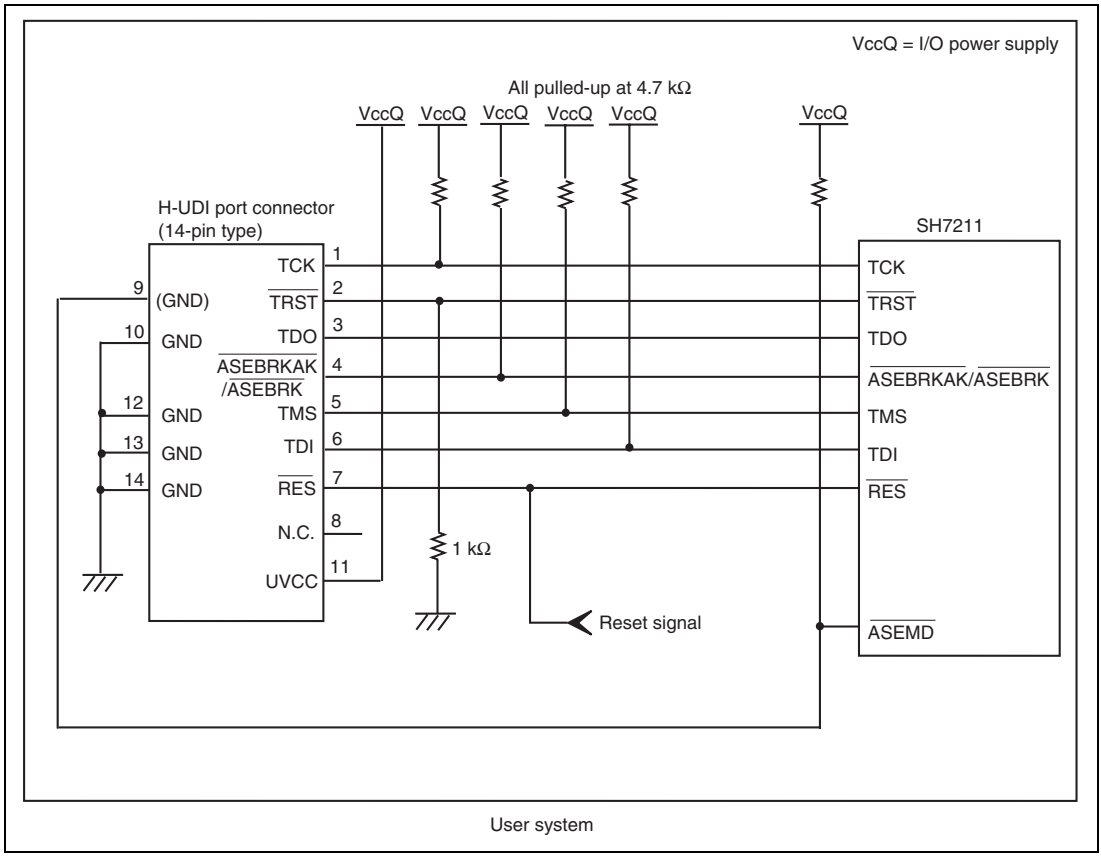


Figure 1.6 Recommended Circuit for Connection between the H-UDI Port Connector and MCU when the Emulator is in Use (14-Pin Type)

1.5.3 Recommended Circuit (38-Pin Type)

Figure 1.7 shows a recommended circuit for connection between the H-UDI and AUD port connectors (38 pins) and the MCU when the emulator is in use.

Notes: 1. Do not connect anything to the N.C. pins of the H-UDI port connector.

2. The _ASEMD pin must be 0 when the emulator is connected and 1 when the emulator is not connected, respectively.
   (1) When the emulator is used: _ASEMD = 0
   (2) When the emulator is not used: _ASEMD = 1
   Figure 1.7 shows an example of circuits that allow the _ASEMD pin to be GND (0) whenever the emulator is connected by using the user system interface cable.
   When the _ASEMD pin is changed by switches, etc., ground pin 3. Do not connect this pin to the _ASEMD pin.
3. When a network resistance is used for pull-up, it may be affected by a noise. Separate TCK from other resistances.
4. The pattern between the H-UDI port connector and the MCU must be as short as possible. Do not connect the signal lines to other components on the board.
5. The AUD signals (AUDCK, AUDATA3 to AUDATA0, and _AUDSYNC) operate in high speed. Isometric connection is needed if possible. Do not separate connection nor connect other signal lines adjacently.
6. Since the H-UDI and the AUD of the MCU operate with the PVcc, supply only the PVcc to the UVCC pin. Make the emulator's switch settings so that the user power will be supplied (SW2 = 1 and SW3 = 1).
7. The resistance value shown in figure 1.7 is for reference.
8. For the AUDCK pin, guard the pattern between the H-UDI port connector and the MCU at GND level.
9. The _TRST pin must be at the low level for a certain period when the power is supplied whether the H-UDI is used or not.
10. The GND bus lead at the center of the H-UDI port connector must be grounded.
11. For the pin processing in cases where the emulator is not used, refer to the hardware manual of the related MCU.

When the circuit is connected as shown in figure 1.7, the switches of the emulator are set as SW2 = 1 and SW3 = 1. For details, refer to section 3.8, Setting the DIP Switches, in the SuperH™ Family E10A-USB Emulator User's Manual.

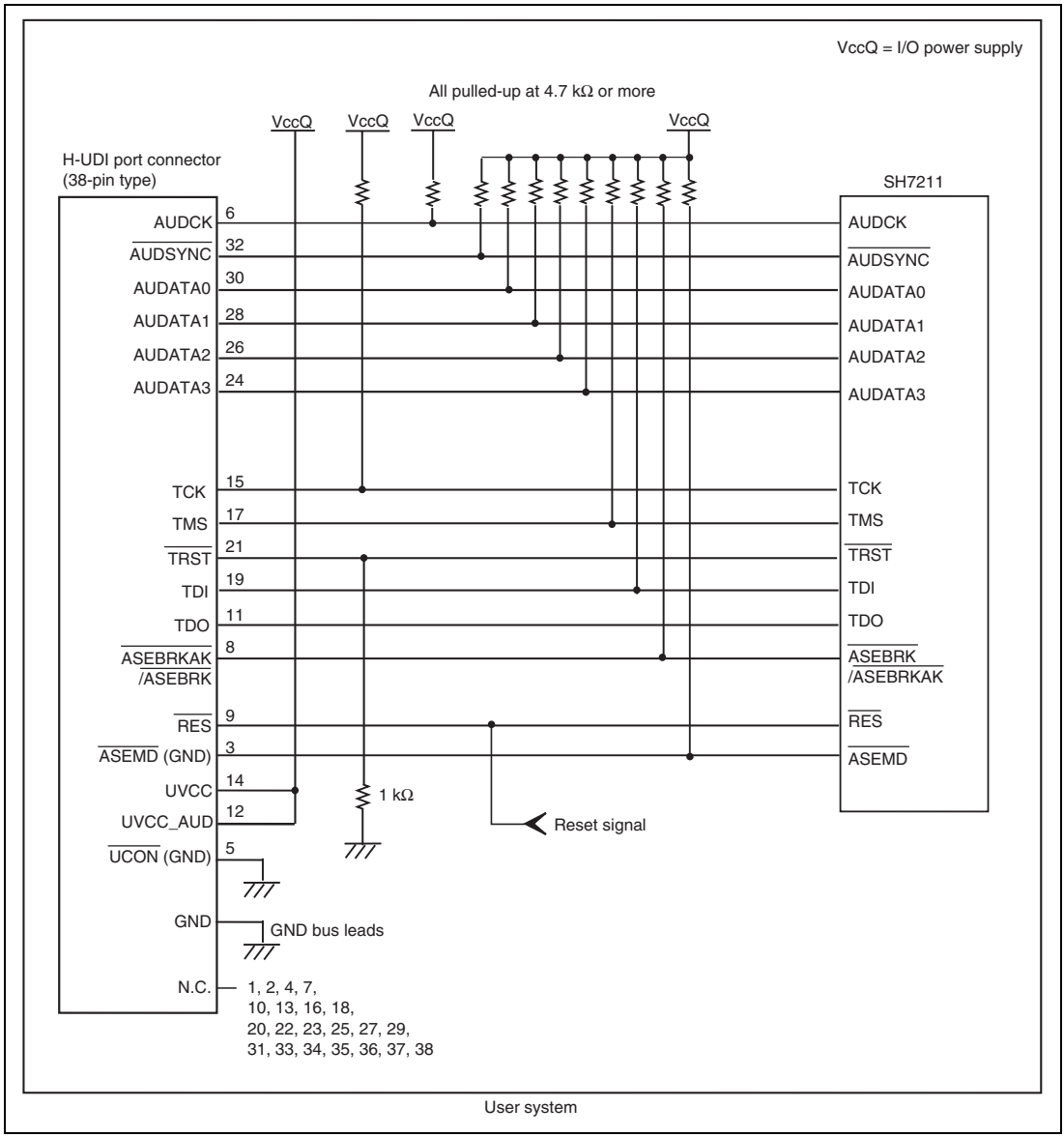


Figure 1.7 Recommended Circuit for Connection between the H-UDI Port Connector and MCU when the Emulator is in Use (38-Pin Type)

Section 2 Software Specifications when Using the SH7211

2.1 Differences between the MCU and the Emulator

1. When the emulator system is initiated, it initializes the general registers and part of the control registers as shown in table 2.1. The initial values of the MCU are undefined. When the emulator is initiated from the workspace, a value to be entered is saved in a session.

Table 2.1 Register Initial Values at Emulator Link Up

Register	Emulator at Link Up
R0 to R14	H'00000000
R15 (SP)	Value of the SP in the power-on reset vector table
PC	Value of the PC in the power-on reset vector table
SR	H'000000F0
GBR	H'00000000
VBR	H'00000000
TBR	H'00000000
MACH	H'00000000
MACL	H'00000000
PR	H'00000000

Note: When a value of the interrupt mask bit in the SR register is changed in the [Registers] window, it is actually reflected in that register immediately before execution of the user program is started. It also applies when the value is changed by the REGISTER_SET command.

2. The emulator uses the H-UDI; do not access the H-UDI.
3. Low-Power States
   - When the emulator is used, the sleep state can be cleared with either the clearing function or with the [STOP] button, and a break will occur.
   - The memory must not be accessed or modified in software standby state.
   - Do not stop inputting the clock to the H-UDI module by using the module standby function.

4. Reset Signals

   The MCU reset signals are only valid during emulation started with clicking the GO or STEP-type button. If these signals are enabled on the user system in command input wait state, they are not sent to the MCU.

Note: Do not break the user program when the _RES, _BREQ, or _WAIT signal is being low. A TIMEOUT error will occur. If the _BREQ or _WAIT signal is fixed to low during break, a TIMEOUT error will occur at memory access.

5. Direct Memory Access Controller (DMAC)

   The DMAC operates even when the emulator is used. When a data transfer request is generated, the DMAC executes DMA transfer.

6. Memory Access during User Program Execution

   During execution of the user program, memory is accessed by the following two methods, as shown in table 2.2.

Table 2.2 Memory Access during User Program Execution

Method	Description
H-UDI read/write	The stopping time of the user program is short because memory is accessed by the dedicated bus master.
Short break	The stopping time of the user program is long because the user program temporarily breaks.

The method for accessing memory during execution of the user program is specified by using the [Configuration] dialog box.

Table 2.3 Stopping Time by Memory Access (Reference)

Method	Condition	Stopping Time
H-UDI read/write	Reading of one longword for the internal RAM	Reading: Maximum three bus clocks (Bϕ)
	Writing of one longword for the internal RAM	Writing: Maximum two bus clocks (Bϕ)
Short break	CPU clock: 160 MHz JTAG clock: 20 MHz Reading or writing of one longword for the external area	About 50 ms

7. Memory Access to the External Flash Memory Area

   The emulator can download the load module to the external flash memory area (for details, refer to section 6.22, Download Function to the Flash Memory Area, in the SuperH™ Family E10A-USB Emulator User's Manual). Other memory write operations are enabled for the RAM area. Therefore, an operation such as memory write or BREAKPOINT should be set only for the RAM area.

8. ROM Cache

   For ROM cache in the MCU, the emulator operates as shown in table 2.4.

Table 2.4 Operation for ROM Cache

Function	Operation
Write and erase	Writes or erases all contents of ROM cache.
Memory read	Accesses the disabled cache area to read the content of internal flash memory.

9. Multiplexing the Emulator Pins

   A part of the emulator pins is multiplexed as shown in table 2.5.

Table 2.5 Multiplexed Functions

Function 1	Function 2
WSCK	TCK
WTXD	TMS
WRXD	_TRST
PB30/_UBCTRG/_IRQOUT/_REFOUT	_ASEBRKAK/_ASEBRK
PB22/RXD2/TCLKD/DACK2/FRAME	_AUDSYNC
PB23/TXD2/TCLKC/DREQ2	AUDCK
PB24/RXD3/TCLKB/IRQ2/TEND1	AUDATA3
PB25/TXD3/TCLKA/IRQ3/DACK1	AUDATA2
PB26/SCK3/TIOC2B/DREQ1	AUDATA1
PB27/TXD3/TIOC2A/TEND0	AUDATA0

When the emulator is connected, function 1 cannot be used because the emulator uses the pins that TCK, TMS, _TRST, and _ASEBRKAK/_ASEBRK have been multiplexed. Function 1 can be used when the pins that AUD has been multiplexed are not connected to the emulator.

The AUD pins are multiplexed with other pins. The AUD function cannot be used for the initial values because they are used as other functions. To use the initial value as the AUD function, set the AUD pins to be used from [AUD pin select] of the [Configuration] dialog box. The emulator rewrites the registers for the pin function controller (PFC) to enable the specified AUD pins before executing the user program. When those registers are changed by the user program, note that the settings of the AUD pins will not be changed.

Table 2.6 shows the bits and the values corresponding to the AUD function.

Table 2.6 Registers and Values Set for the AUD Function

Pin Name of the Port Function	AUD Function	Register and Bit to be Set	Value to be Set
PB23	AUDCK	PBCRH2[10:8]	3'b110
PB22	_AUDSYNC	PBCRH2[10:8]	3'b110
PB24	AUDATA3	PBCRH2[10:8]	3'b110
PB25	AUDATA2	PBCRH2[10:8]	3'b110
PB26	AUDATA1	PBCRH2[10:8]	3'b110
PB27	AUDATA0	PBCRH2[10:8]	3'b110

10. Using WDT

    The WDT does not operate during break.

11. Loading Sessions

    Information in [JTAG clock] of the [Configuration] dialog box cannot be recovered by loading sessions. Thus the TCK value will be as follows:

    - When HS0005KCU01H or HS0005KCU02H is used: TCK = 2.5 MHz

12. [IO] Window

    - Display and modification

      For each watchdog timer register, there are two registers to be separately used for write and read operations.

Table 2.7 Watchdog Timer Register

Register Name	Usage	Register
WTCSR(W)	Write	Watchdog timer control/status register
WTCNT(W)	Write	Watchdog timer counter
WTCSR(R)	Read	Watchdog timer control/status register
WTCNT(R)	Read	Watchdog timer counter
WRCSR(W)	Write	Watchdog reset control/status register
WRCSR(R)	Read	Watchdog reset control/status register

- Customization of the I/O-register definition file

  The internal I/O registers can be accessed from the [IO] window. However, note the following when accessing the SDMR register of the bus-state controller. Before accessing the SDMR register, specify addresses to be accessed in the I/O-register definition file (SH7211.IO) and then activate the High-performance Embedded Workshop. After the I/O-register definition file is created, the MCU's specifications may be changed. If each I/O register in the I/O-register definition file differs from addresses described in the hardware manual, change the I/O-register definition file according to the description in the hardware manual. The I/O-register definition file can be customized depending on its format. However, the emulator does not support the bit-field function.
- Verify

  In the [IO] window, the verify function of the input value is disabled.

13. Illegal Instructions

    Do not execute illegal instructions with STEP-type commands.

14. Reset Input

    During execution of the user program, the emulator may not operate correctly if a contention occurs between the following operations for the emulator and the reset input to the target device:

    - Setting an Event Condition
    - Setting an internal trace
    - Displaying the content acquired by an internal trace
    - Reading or writing of a memory

    Note that those operations should not contend with the reset input to the target device.

15. Contention between the Change of the FRQCR Register and the Debugging Functions

The following notes are required for the user program for changing the multiplication rate of PLL circuit 1 to change the frequency:

- Avoid contention between the change of the FRQCR register in the user program and the memory access from the [Memory] window, etc.
- When the automatic updating function is used in the [Monitor] window or [Watch] window, generate and set a break of Event Condition for an instruction immediately before changing the FRQCR register. Contention will be avoided by generating a break and executing the user program again.

For the change of the multiplication rate of PLL circuit 1 and the FRQCR register, refer to the hardware manual for the MCU.

2.2 Specific Functions for the Emulator when Using the SH7211

2.2.1 Event Condition Functions

The emulator is used to set event conditions for the following three functions:

- Break of the user program
- Internal trace
- Start or end of performance measurement

Table 2.8 lists the types of Event Condition.

Table 2.8 Types of Event Condition

Event Condition Type	Description
Address bus condition (Address)	Sets a condition when the address bus (data access) value or the program counter value (before or after execution of instructions) is matched.
Data bus condition (Data)	Sets a condition when the data bus value is matched. Byte, word, or longword can be specified as the access data size.
Bus state condition (Bus State)	There are two bus state condition settings: Bus state condition: Sets a condition when the data bus value is matched. Read/Write condition: Sets a condition when the read/write condition is matched.
Count	Sets a condition when the specified other conditions are satisfied for the specified counts.
Reset point	A reset point is set when the count and the sequential condition are specified.
Action	Selects the operation when a condition (such as a break, a trace halt condition, or a trace acquisition condition) is matched.

Using the [Combination action (Sequential or PtoP)] dialog box, which is opened by selecting [Combination action (Sequential or PtoP)] from the pop-up menu on the [Event Condition] sheet, specifies the sequential condition and the start or end of performance measurement.

Table 2.9 lists the combinations of conditions that can be set under Ch1 to Ch11 and the software trace.

Table 2.9 Dialog Boxes for Setting Event Conditions

Dialog Box		Function: Address Bus Condition (Address)	Function: Data Bus Condition (Data)	Function: Bus State Condition (Bus Status)	Function: Count Condition (Count)	Function: Action
[Event Condition 1]	Ch1	O	O	O	O	O (B, T1, and P)
[Event Condition 2]	Ch2	O	O	O	X	O (B, T1, and P)
[Event Condition 3]	Ch3	O	X	X	X	O (B and T2)
[Event Condition 4]	Ch4	O	X	X	X	O (B and T3)
[Event Condition 5]	Ch5	O	X	X	X	O (B and T3)
[Event Condition 6]	Ch6	O	X	X	X	O (B and T2)
[Event Condition 7]	Ch7	O	X	X	X	O (B and T2)
[Event Condition 8]	Ch8	O	X	X	X	O (B and T2)
[Event Condition 9]	Ch9	O	X	X	X	O (B and T2)
[Event Condition 10]	Ch10	O	X	X	X	O (B and T2)
[Event Condition 11]	Ch11	O (reset point)	X	X	X	O (B and T2)

Notes: 1. O: Can be set in the dialog box.
X: Cannot be set in the dialog box.

2. For the Action item,
B: Setting a break is enabled.
T1: Setting the trace halt and acquisition conditions are enabled for the internal trace.
T2: Setting the trace halt is enabled for the internal trace.
T3: Setting the trace halt and point-to-point is enabled for the internal trace.
P: Setting a performance-measurement start or end condition is enabled.

The [Event Condition 11] dialog box is used to specify the count of [Event Condition 1] and becomes a reset point when the sequential condition is specified.

Sequential Setting: Using the [Combination action (Sequential or PtoP)] dialog box specifies the sequential condition and the start or end of performance measurement.

Table 2.10 Conditions to Be Set

Classification	Item	Description
[Ch1, 2, 3] list box	Sets the sequential condition and the start or end of performance measurement using Event Conditions 1 to 3 and 11.	
	Don't care	Sets no sequential condition or the start or end of performance measurement.
	Break: Ch3-2-1	Breaks when a condition is satisfied in the order of Event Condition 3, 2, 1.
	Break: Ch3-2-1, Reset point	Breaks when a condition is satisfied in the order of Event Condition 3, 2, 1. Enables the reset point of Event Condition 11.
	Break: Ch2-1	Breaks when a condition is satisfied in the order of Event Condition 2, 1.
	Break: Ch2-1, Reset point	Breaks when a condition is satisfied in the order of Event Condition 2, 1. Enables the reset point.
	I-Trace stop: Ch3-2-1	Halts acquisition of an internal trace when a condition is satisfied in the order of Event Condition 3, 2, 1.
	I-Trace stop: Ch3-2-1, Reset point	Halts acquisition of an internal trace when a condition is satisfied in the order of Event Condition 3, 2, 1. Enables the reset point.
	I-Trace stop: Ch2-1	Halts acquisition of an internal trace when a condition is satisfied in the order of Event Condition 2, 1.
	I-Trace stop: Ch2-1, Reset point	Halts acquisition of an internal trace when a condition is satisfied in the order of Event Condition 2, 1. Enables the reset point.
	Ch2 to Ch1 PA	Sets the performance measurement period during the time from the satisfaction of the condition set in Event Condition 2 (start condition) to the satisfaction of the condition set in Event Condition 1 (end condition).

Table 2.10 Conditions to Be Set (cont)

Classification	Item	Description
[Ch1, 2, 3] list box (cont)	Ch1 to Ch2 PA	Sets the performance measurement period during the time from the satisfaction of the condition set in Event Condition 1 (start condition) to the satisfaction of the condition set in Event Condition 2 (end condition).
[Ch4, 5] list box	Sets the point-to-point of the internal trace (the start or end condition of trace acquisition) using Event Conditions 4 and 5.	
	Don't care	Sets no start or end condition of trace acquisition.
	I-Trace: Ch5 to Ch4 PtoP	Sets the acquisition period during the time from the satisfaction of the condition set in Event Condition 5 (start condition) to the satisfaction of the condition set in Event Condition 4 (end condition).
	I-Trace: Ch5 to Ch4 PtoP, power-on reset	Sets the acquisition period during the time from the satisfaction of the condition set in Event Condition 5 (start condition) to the satisfaction of the condition set in Event Condition 4 (end condition) or the power-on reset.

Notes:
1. After the sequential condition and the count specification condition of Event Condition 1 have been set, break and trace acquisition will be halted if the sequential condition is satisfied for the specified count.
2. If a reset point is satisfied, the satisfaction of the condition set in Event Condition will be disabled. For example, if the condition is satisfied in the order of Event Condition 3, 2, reset point, 1, the break or trace acquisition will not be halted. If the condition is satisfied in the order of Event Condition 3, 2, reset point, 3, 2, 1, the break and trace acquisition will be halted.
3. If the start condition is satisfied after the end condition has been satisfied by measuring performance, performance measurement will be restarted. For the measurement result after a break, the measurement results during performance measurement are added.
4. If the start condition is satisfied after the end condition has been satisfied by the point-to-point of the internal trace, trace acquisition will be restarted.

Usage Example of Sequential Break Extension Setting: A tutorial program provided for the product is used as an example. For the tutorial program, refer to section 6, Tutorial, in the SuperH™ Family E10A-USB Emulator User's Manual.

The conditions of Event Condition are set as follows:

1. Ch3

   Breaks address H'00001068 when the condition [Only program fetched address after] is satisfied.
2. Ch2

   Breaks address H'0000107a when the condition [Only program fetched address after] is satisfied.
3. Ch1

   Breaks address H'00001086 when the condition [Only program fetched address after] is satisfied.

   Note: Do not set other channels.
4. Sets the content of the [Ch1,2,3] list box to [Break: Ch 3-2-1] in the [Combination action (Sequential or PtoP)] dialog box.
5. Enables the condition of Event Condition 1 from the popup menu by clicking the right mouse button on the [Event Condition] sheet.

Then, set the program counter and stack pointer (PC = H'00000800, R15 = H'00010000) in the [Registers] window and click the [Go] button. If this does not execute normally, issue a reset and execute the above procedures.

The program is executed up to the condition of Ch1 and halted. Here, the condition is satisfied in the order of Ch3 -> 2 -> 1.

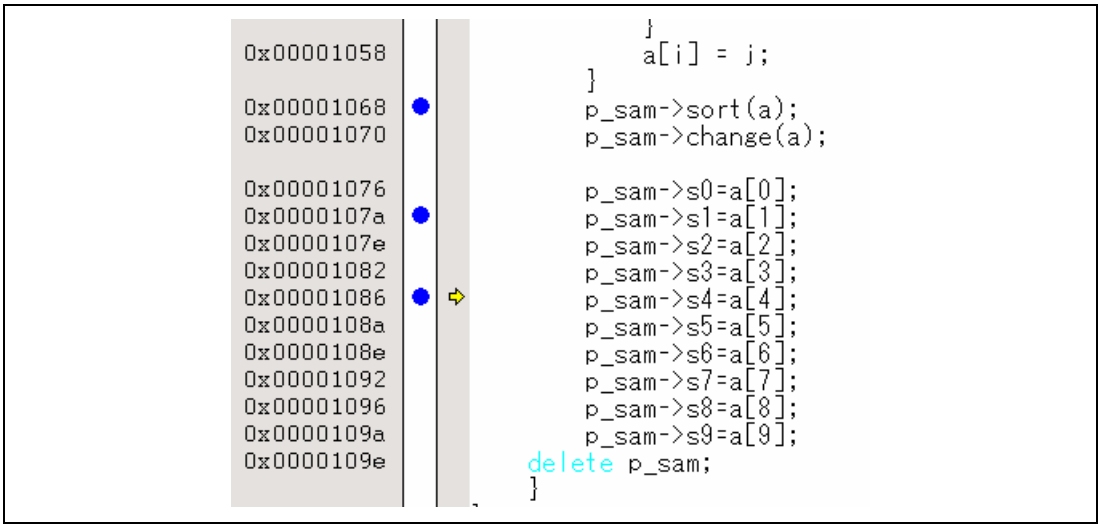


Figure 2.1 [Source] Window at Execution Halted (Sequential Break)

If the sequential condition, performance measurement start/end, or point-to-point for the internal trace is set, conditions of Event Condition to be used will be disabled. Such conditions must be enabled from the popup menu by clicking the right mouse button on the [Event Condition] sheet.

Notes:
1. If the Event condition is set for the slot in the delayed branch instruction by the program counter (after execution of the instruction), the condition is satisfied before executing the instruction in the branch destination (when a break has been set, it occurs before executing the instruction in the branch destination).
2. Do not set the Event condition for the SLEEP instruction by the program counter (after execution of the instruction).
3. When the Event condition is set for the 32-bit instruction by the program counter, set that condition in the upper 16 bits of the instruction.
4. If the power-on reset and the Event condition are matched simultaneously, no condition will be satisfied.
5. Do not set the Event condition for the DIVU or DIVS instruction by the program counter (after execution of the instruction).
6. If a condition of which intervals are satisfied closely is set, no sequential condition will be satisfied.
   - Set the Event conditions, which are satisfied closely, by the program counter with intervals of two or more instructions.
   - After the Event condition has been matched by accessing data, set the Event condition by the program counter with intervals of 17 or more instructions.

7. If the settings of the Event condition or the sequential conditions are changed during execution of the program, execution will be suspended. (The number of clocks to be suspended during execution of the program is a maximum of about 102 bus clocks (Bϕ). If the bus clock (Bϕ) is 66.6 MHz, the program will be suspended for 1.53 μs.)
8. If the settings of Event conditions or the sequential conditions are changed during execution of the program, the emulator temporarily disables all Event conditions to change the settings. During this period, no Event condition will be satisfied.
9. If the break condition before executing an instruction is set to the instruction followed by DIVU and DIVS, the factor for halting a break will be incorrect under the following condition:
   If a break occurs during execution of the above DIVU and DIVS instructions, the break condition before executing an instruction, which has been set to the next instruction, may be displayed as the factor for halting a break.
10. If the break conditions before and after executing instructions are set to the same address, the factor for halting a break will be incorrectly displayed. The factor for halting a break due to the break condition after executing an instruction will be displayed even if a break is halted by the break condition before executing an instruction.
11. Do not set the break condition after executing instructions and BREAKPOINT (software break) to the same address.
12. When the emulator is being connected, the user break controller (UBC) function is not available.

2.2.2 Trace Functions

The emulator supports the trace functions listed in table 2.11.

Table 2.11 Trace Functions

Function	Internal Trace	AUD Trace
Branch trace	Supported	Supported
Memory access trace	Supported	Supported
Software trace	Not supported	Supported

Table 2.12 shows the type numbers that the AUD function can be used.

Table 2.12 Type Number and AUD Function

Type Number	AUD Function
HS0005KCU01H	Not supported
HS0005KCU02H	Supported

The internal and AUD traces are set in the [Acquisition] dialog box of the [Trace] window.

Internal Trace Function: When [I-Trace] is selected for [Trace type] on the [Trace Mode] page of the [Acquisition] dialog box, the internal trace can be used.

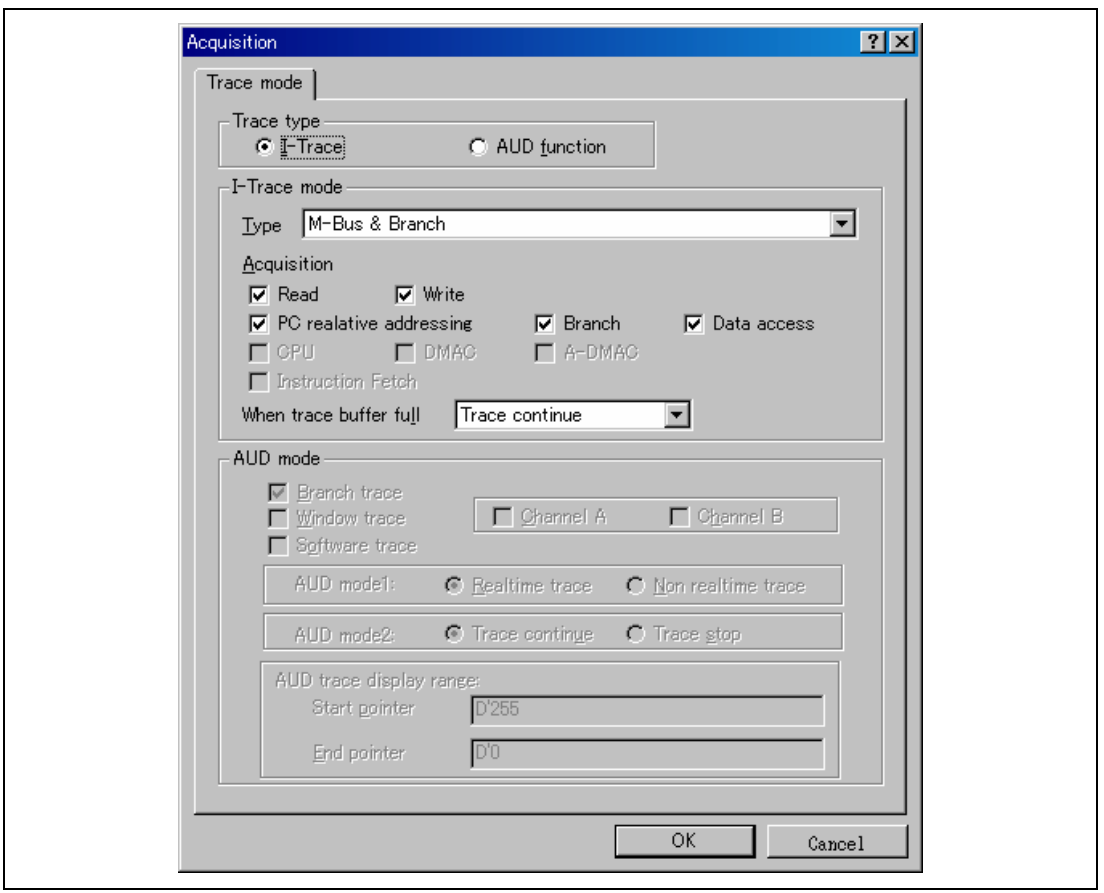


Figure 2.2 [Acquisition] Dialog Box – Internal Trace Function

The following three items can be selected as the internal trace from [Type] of [I-Trace mode].

Table 2.13 Information on Acquiring the Internal Trace

Item	Acquisition Information
[M-Bus & Branch]	Acquires the data and branch information on the M-bus. • Data access (read/write) • PC-relative access • Branch information
[I-Bus]	Acquires the data on the I-bus. • Data access (read/write) • Selection of the bus master on the I-bus (CPU/DMA/A-DMA) • Instruction fetch
[I-Bus, M-Bus & Branch]	Acquires the contents of [M-Bus & Branch] and [I-Bus].

After selecting [Type] of [I-Trace mode], select the content to be acquired from [Acquisition]. Typical examples are described below (note that items disabled for [Acquisition] are not acquired).

- Example of acquiring branch information only:
  Select [M-Bus & Branch] from [Type] and enable [Branch] on [Acquisition].
- Example of acquiring the read or write access (M-bus) only by a user program:
  Select [M-Bus & Branch] from [Type] and enable [Read], [Write], and [Data access] on [Acquisition].
- Example of acquiring the read access only by DMA (I-bus):
  Select [I-Bus] from [Type] and enable [Read], [DMA], and [Data access] on [Acquisition].

Using Event Condition restricts the condition; the following three items are set as the internal trace conditions.

Table 2.14 Trace Conditions of the Internal Trace

Item	Acquisition Information
Trace halt	Acquires the internal trace until the Event Condition is satisfied. (The trace content is displayed in the [Trace] window after a trace has been halted. No break occurs in the user program.)
Trace acquisition	Acquires only the data access where the Event Condition is satisfied.
Point-to-point	Traces the period from the satisfaction of Event Condition 4 to the satisfaction of Event Condition 5.

To restrict trace acquisition to access for only a specific address or specific function of a program, an Event Condition can be used. Typical examples are described below.

- Example of halting a trace with a write access (M-bus) to H'FFF80000 by the user program as a condition (trace halt):

  Set the condition to be acquired on [I-Trace mode].

  Set the following in the [Event Condition 1] or [Event Condition 2] dialog box:

  Address condition: Set [Address] and H'FFF80000.

  Bus state condition: Set [M-Bus] and [Write].

  Action condition: Disable [Acquire Break] and set [Acquire Trace] for [Stop].

- Example of acquiring the write access (M-bus) only to H'FFF80000 by the user program (trace acquisition condition):

  Select [M-Bus & Branch] from [Type] and enable [Write] and [Data access] on [Acquisition].

  Set the following in the [Event Condition 1] or [Event Condition 2] dialog box:

  Address condition: Set [Address] and H'FFF80000.

  Bus state condition: Set [M-Bus] and [Write].

  Action condition: Disable [Acquire Break] and set [Acquire Trace] for [Condition].

For the trace acquisition condition, the condition to be acquired by Event Condition should be acquired by [I-Trace mode].

- Example of acquiring a trace for the period while the program passes H'1000 through H'2000 (point-to-point):

  Set the condition to be acquired on [I-Trace mode].

  Set the address condition as H'1000 in the [Event Condition 5] dialog box.

  Set the address condition as H'2000 in the [Event Condition 4] dialog box.

  Set [I-Trace] as [Ch5 to Ch4 PtoP] in the [Combination action (Sequential or PtoP)] dialog box.

When point-to-point and trace acquisition condition are set simultaneously, they are ANDed.

Notes on Internal Trace:

- Timestamp

  The timestamp is the clock counts of Bϕ (48-bit counter). Table 2.15 shows the timing for acquiring the timestamp.

Table 2.15 Timing for the Timestamp Acquisition

Item	Acquisition Information	Counter Value Stored in the Trace Memory
M-bus data access		Counter value when data access (read or write) has been completed
Branch		Counter value when the next bus cycle has been completed after a branch
I-bus	Fetch	Counter value when a fetch has been completed
	Data access	Counter value when data access has been completed

- Point-to-point

  The trace-start condition is satisfied when the specified instruction has been fetched. Accordingly, if the trace-start condition has been set for the overrun-fetched instruction (an instruction that is not executed although it has been fetched at a branch or transition to an interrupt), tracing is started during overrun-fetching of the instruction. However, when overrun-fetching is achieved (a branch is completed), tracing is automatically suspended. If the start and end conditions are satisfied closely, trace information will not be acquired correctly.

  The execution cycle of the instruction fetched before the start condition is satisfied may be traced.

  When the I-bus is acquired, do not specify point-to-point.

  Memory access may not be acquired by the internal trace if it occurs at several instructions immediately before satisfaction of the point-to-point end condition.

- Halting a trace

  Do not set the trace end condition for the sleep instruction and the branch instruction that the delay slot becomes the sleep instruction.

- Trace acquisition condition

  Do not set the trace end condition for the sleep instruction and the branch instruction according to which the delay slot becomes the sleep instruction.
  When [I-Bus, M-Bus & Branch] is selected and the trace acquisition condition is set for the M-bus and I-bus with Event Condition, set the M-bus condition and the I-bus condition for [Event Condition 1] and [Event Condition 2], respectively.
  If the settings of [I-Trace mode] are changed during execution of the program, execution will be suspended. (The number of clocks to be suspended during execution of the program is a maximum of about 51 peripheral clocks (Pϕ) + 4096 bus clocks (Bϕ). If the peripheral clock (Pϕ) is 33.3 MHz and the bus clock (Bϕ) is 66.6 MHz, the program will be suspended for 1.757 μs.)

- Displaying a trace

  If a trace is displayed during execution of the program, execution will be suspended to acquire the trace information. (The number of clocks to be suspended during execution of the program is a maximum of about 20480 peripheral clocks (Pϕ) + 4096 bus clocks (Bϕ). If the peripheral clock (Pϕ) is 33.3 MHz and the bus clock (Bϕ) is 66.6 MHz, the program will be suspended for 676.52 μs.)

- Branch trace

  If breaks occur immediately after executing non-delayed branch and TRAPA instructions and generating a branch due to exception or interrupt, a trace for one branch will not be acquired immediately before such breaks.
  However, this does not affect on generation of breaks caused by a BREAKPOINT and a break before executing instructions of Event Condition.

- Writing memory immediately before generating a break

  If an instruction is executed to write memory immediately before generating a break, trace acquisition may not be performed.

AUD Trace Functions: This function is operational when the AUD pin of the device is connected to the emulator. Table 2.16 shows the AUD trace acquisition mode that can be set in each trace function.

Table 2.16 AUD Trace Acquisition Mode

Type	Mode	Description
Continuous trace occurs	Realtime trace	When the next branch occurs while the trace information is being output, all the information may not be output. The user program can be executed in realtime, but some trace information will be lost.
	Non realtime trace	When the next branch occurs while the trace information is being output, the CPU stops operations until the information is output. The user program is not executed in realtime.
Trace buffer full	Trace continue	This function overwrites the latest trace information to store the oldest trace information.
	Trace stop	After the trace buffer becomes full, the trace information is no longer acquired. The user program is continuously executed.

To set the AUD trace acquisition mode, click the [Trace] window with the right mouse button and select [Setting] from the pop-up menu to display the [Acquisition] dialog box. The AUD trace acquisition mode can be set in the [AUD mode1] or [AUD mode2] group box in the [Trace mode] page of the [Acquisition] dialog box.

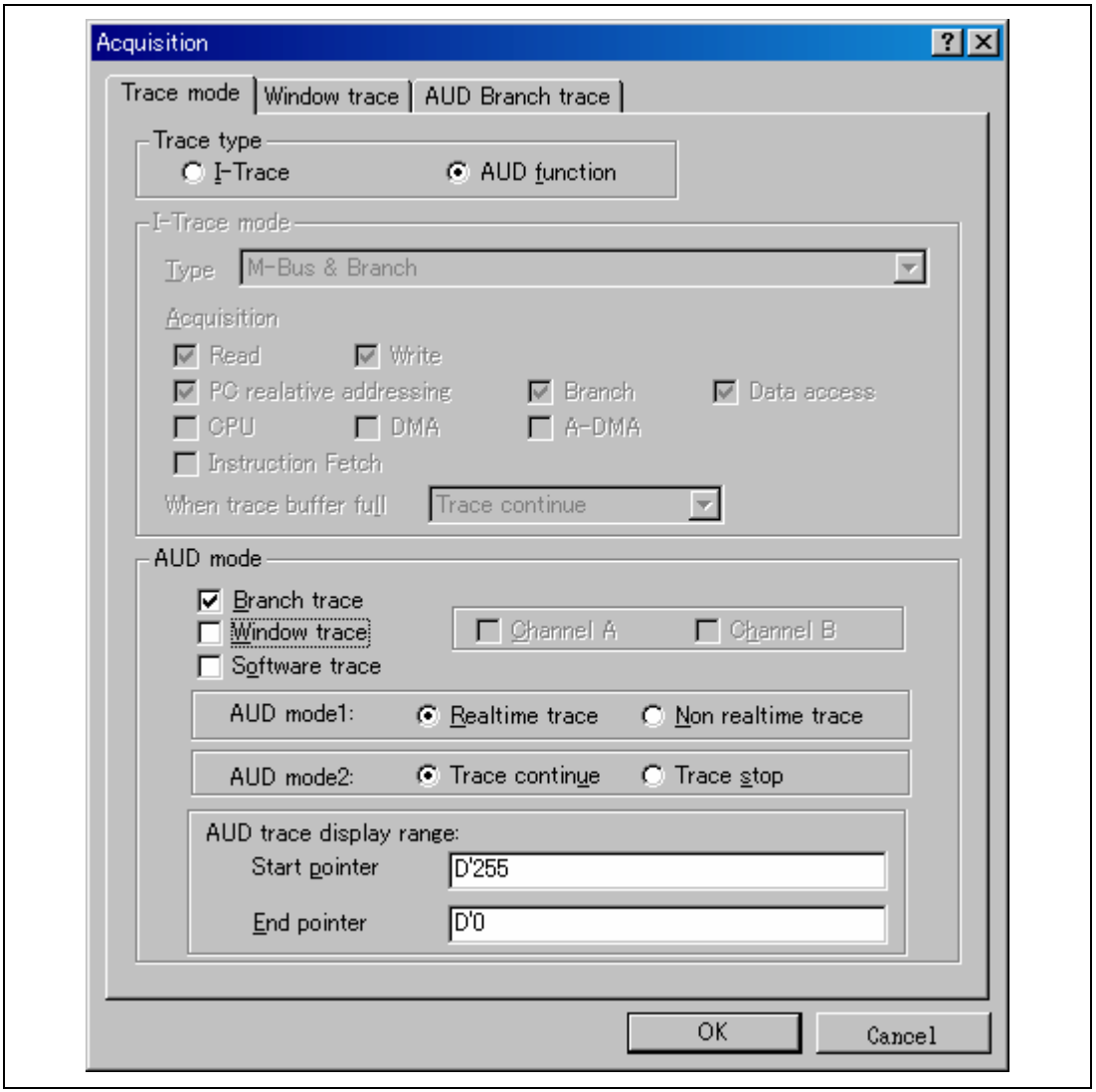


Figure 2.3 [Trace mode] Page

When the AUD trace function is used, select the [AUD function] radio button in the [Trace type] group box of the [Trace mode] page.

(a) Branch Trace Function

The branch source and destination addresses and their source lines are displayed.

Branch trace can be acquired by selecting the [Branch trace] check box in the [AUD function] group box of the [Trace mode] page.

The branch type can be selected in the [AUD Branch trace] page.

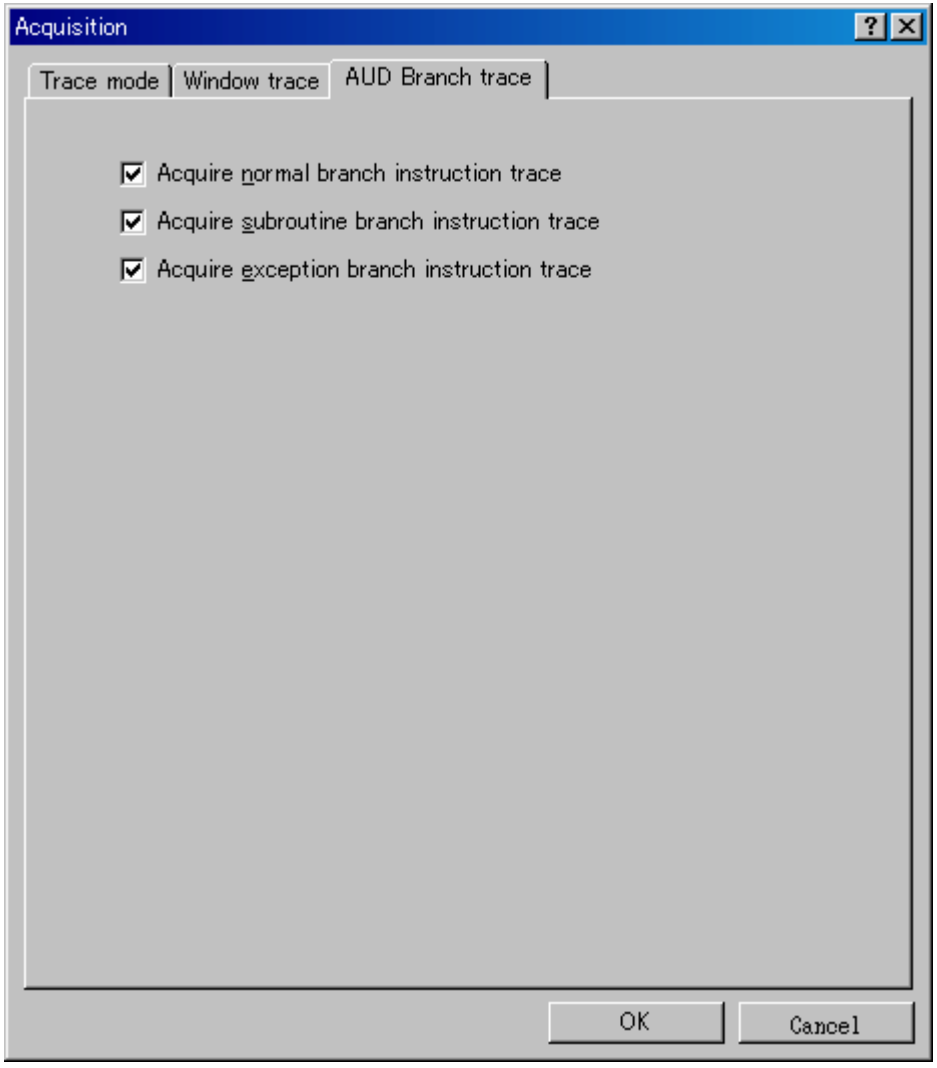


Figure 2.4 [AUD Branch trace] Page

(b) Window Trace Function

Memory access in the specified range can be acquired by trace.

Two memory ranges can be specified for channels A and B. The read, write, or read/write cycle can be selected as the bus cycle for trace acquisition.

[Setting Method]

(i) Select the [Channel A] and [Channel B] check boxes in the [AUD function] group box of the [Trace mode] page. Each channel will become valid.

(ii) Open the [Window trace] page and specify the bus cycle, memory range, and bus type that are to be set for each channel.

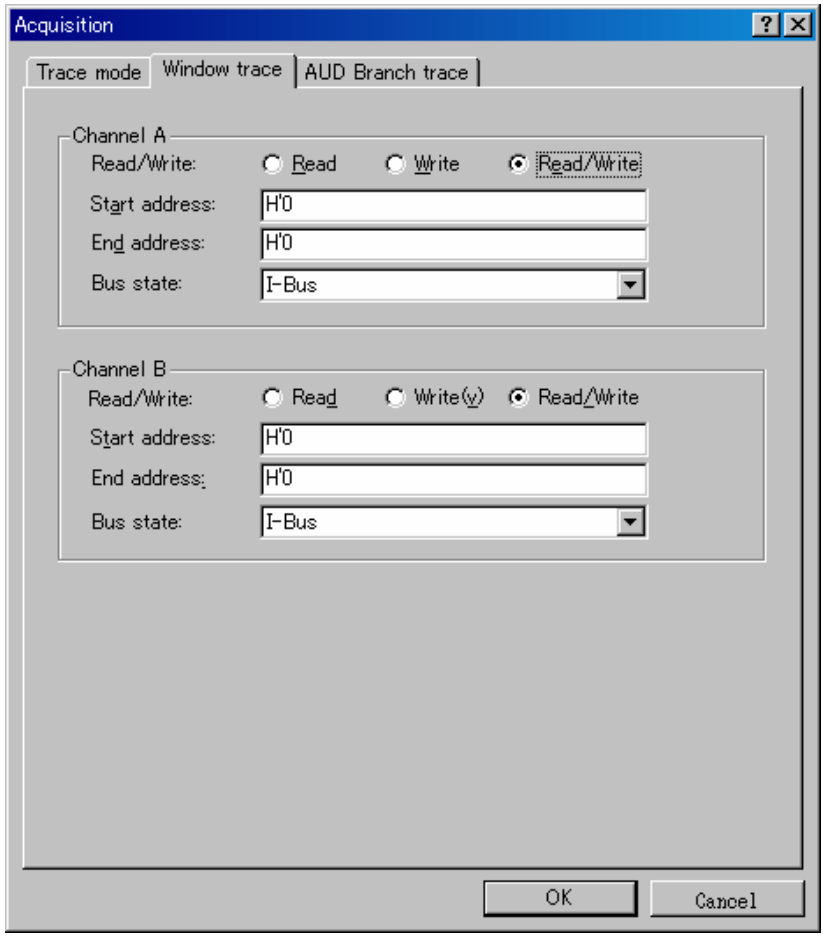


Figure 2.5 [Window trace] Page

Note: When [M-Bus] or [I-Bus] is selected, the following bus cycles will be traced.

- M-Bus: A bus cycle generated by the CPU is acquired.
- I-Bus: A bus cycle generated by the CPU or DMA is acquired.

(c) Software Trace Function

Note: This function can be supported with SHC/C++ compiler (manufactured by Renesas Technology Corp.; including OEM and bundle products) V7.0 or later.

When a specific instruction is executed, the PC value at execution and the contents of one general register are acquired by trace. Describe the Trace(x) function (x is a variable name) to be compiled and linked beforehand. For details, refer to the SHC manual.

When the load module is downloaded on the target system and is executed while a software trace function is valid, the PC value that has executed the Trace(x) function, the general register value for x, and the source lines are displayed.

To activate the software trace function, select the [Software trace] check box in the [AUD function] group box of the [Trace mode] page.

Notes on AUD Trace:

1. When the trace display is performed during user program execution, the mnemonics, operands, or source is not displayed.
2. The AUD trace function outputs the differences between newly output branch source addresses and previously output branch source addresses. The window trace function outputs the differences between newly output addresses and previously output addresses. If the previous branch source address is the same as the upper 16 bits, the lower 16 bits are output. If it matches the upper 24 bits, the lower 8 bits are output. If it matches the upper 28 bits, the lower 4 bits are output.
   The emulator regenerates the 32-bit address from these differences and displays it in the [Trace] window. If the emulator cannot display the 32-bit address, it displays the difference from the previously displayed 32-bit address.
3. If the 32-bit address cannot be displayed, the source line is not displayed.
4. If a completion-type exception occurs during exception branch acquisition, the next address to the address in which an exception occurs is acquired.
5. The AUD trace is disabled while the profiling function is used.
6. Set the AUD clock (AUDCK) frequency to 40 MHz or lower. If the frequency is higher than 40 MHz, the emulator will not operate normally.
7. If breaks occur immediately after executing non-delayed branch and TRAPA instructions and generating a branch due to exception or interrupt, a trace for one branch will not be acquired immediately before such breaks. However, this does not affect on generation of breaks caused by a BREAKPOINT and a break before executing instructions of Event Condition.
8. For the result by software trace, a value in the [Data] item is not correct (that value is correct for window trace).

2.2.3 Note on Using the JTAG (H-UDI) Clock (TCK)

1. Set the JTAG clock (TCK) frequency to lower than the frequency of the peripheral module clock and to 10 MHz or lower.
2. The initial value of the JTAG clock (TCK) is 2.5 MHz.
3. A value to be set for the JTAG clock (TCK) is initialized after executing [Reset CPU] or [Reset Go]. Thus the TCK value will be 2.5 MHz.

2.2.4 Notes on Setting the [Breakpoint] Dialog Box

1. When an odd address is set, the next lowest even address is used.
2. A BREAKPOINT is accomplished by replacing instructions of the specified address. It cannot be set to the following addresses:
   - An area other than CS and the internal RAM
   - An instruction in which Break Condition 2 is satisfied
   - A slot instruction of a delayed branch instruction
3. During step operation, specifying BREAKPOINTs and Event Condition breaks are disabled.
4. When execution resumes from the address where a BREAKPOINT is specified and a break occurs before Event Condition execution, single-step operation is performed at the address before execution resumes. Therefore, realtime operation cannot be performed.
5. When a BREAKPOINT is set to the slot instruction of a delayed branch instruction, the PC value becomes an illegal value. Accordingly, do not set a BREAKPOINT to the slot instruction of a delayed branch instruction.
6. If an address of a BREAKPOINT cannot be correctly set in the ROM or flash memory area, a mark ● will be displayed in the [BP] area of the address on the [Source] or [Disassembly] window by refreshing the [Memory] window, etc. after Go execution. However, no break will occur at this address. When the program halts with the event condition, the mark ● disappears.
7. If you wish to use a BREAKPOINT (software break), specify the SH2A_SBSTK command to enable use of a user stack before setting a PC break. While enabled, extra four bytes of a user stack are used when a break occurs. The value of the stack pointer (R15) must be correctly set in advance because a user stack is to be used. By default, use of a user stack is disabled. For details on the command, refer to the help file.

- Example
  To enable use of a user stack:
  >SH2A_SBSTK enable

2.2.5 Notes on Setting the [Event Condition] Dialog Box and the BREAKCONDITION_SET Command

1. When [Go to cursor], [Step In], [Step Over], or [Step Out] is selected, the settings of Event Condition 3 are disabled.
2. When an Event Condition is satisfied, emulation may stop after two or more instructions have been executed.

2.2.6 Performance Measurement Function

The emulator supports the performance measurement function.

1. Setting the performance measurement conditions

   To set the performance measurement conditions, use the [Performance Analysis] dialog box and the PERFORMANCE_SET command. When any line in the [Performance Analysis] window is clicked with the right mouse button, a popup menu is displayed and the [Performance Analysis] dialog box can be displayed by selecting [Setting].

Note: For the command line syntax, refer to the online help.

(a) Specifying the measurement start/end conditions

The measurement start/end conditions are specified by using Event Condition 1,2. The [Ch1,2,3] list box of the [Combination action (Sequential or PtoP)] dialog box can be used.

Table 2.17 Measurement Period

Classification	Item	Description
Selection in the [Ch1, 2, 3] list box	Ch2 to Ch1 PA	The period from the satisfaction of the condition set in Event Condition 2 (start condition) to the satisfaction of the condition set in Event Condition 1 (end condition) is set as the performance measurement period.
	Ch1 to Ch2 PA	The period from the satisfaction of the condition set in Event Condition 1 (start condition) to the satisfaction of the condition set in Event Condition 2 (end condition) is set as the performance measurement period.
	Other than above	The period from the start of execution of the user program to the occurrence of a break is measured.

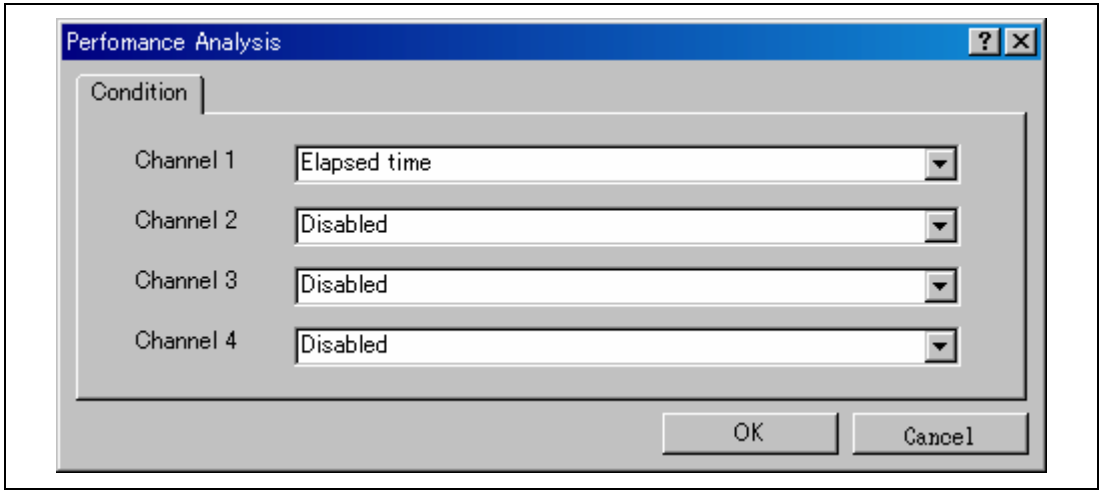


Figure 2.6 [Performance Analysis] Dialog Box

For measurement tolerance,

- The measured value includes tolerance.
- Tolerance will be generated before or after a break.

Note: When [Ch2 to Ch1 PA] or [Ch1 to Ch2 PA] is selected, to execute the user program, specify conditions set in Event Condition 2 and Event Condition 1 and one or more items for performance measurement.

(b) Measurement item

Items are measured with [Channel 1 to 4] in the [Performance Analysis] dialog box. Maximum four conditions can be specified at the same time.

Table 2.18 Measurement Item

Selected Name	Option
Disabled	None
Elapsed time	AC (The number of execution cycles (Iϕ) is set as the measurement item.)
Branch instruction counts	BT
Number of execution instructions	I
Number of execution 32bit-instructions	I32
Exception/interrupt counts	EA
Interrupt counts	INT
Data cache-miss counts	DC
Instruction cache-miss counts	IC
All area access counts	ARN
All area instruction access counts	ARIN
All area data access counts	ARND
Cacheable area access counts	CDN (data access)
Cacheable area instruction access counts	CIN
Non cacheable area data access counts	NCN
URAM area access counts	UN
URAM area instruction access counts	UIN
URAM area data access counts	UDN
Internal I/O area data access counts	IODN
Internal ROM area access counts	RN
Internal ROM area instruction access counts	RIN
Internal ROM area data access counts	RDN
All area access cycle	ARC
All area instruction access cycle	ARIC
All area data access cycle	ARDC
All area access stall	ARS
All area instruction access stall	ARIS
All area data access stall	ARDS

Note: Selected names are displayed for CONDITION in the [Performance Analysis] window. Options are parameters for <mode> of the PERFORMANCE_SET command.

Notes: 1. In the non-realtime trace mode of the AUD trace, normal counting cannot be performed because the generation state of the stall or the execution cycle is changed.
2. If the internal ROM is not installed on the product, do not set the measurement item for the internal ROM area.
3. For SH7211, do not set measurement items for the cache-miss counts, cacheable area, or non-cacheable area.

2. Displaying the measured result

The measured result is displayed in the [Performance Analysis] window or the PERFORMANCE_ANALYSIS command with hexadecimal (32 bits).

Note: If a performance counter overflows as a result of measurement, "********" will be displayed.

3. Initializing the measured result

To initialize the measured result, select [Initialize] from the popup menu in the [Performance Analysis] window or specify INIT with the PERFORMANCE_ANALYSIS command.

SuperH™ Family E10A-USB Emulator
Additional Document for User's Manual
Supplementary Information on Using the SH7211

Publication Date: Rev.1.00, February 28, 2006
Rev.2.00, April 10, 2007
Published by: Sales Strategic Planning Div.
Renesas Technology Corp.
Edited by: Customer Support Department
Global Strategic Communication Div.
Renesas Solutions Corp.

© 2007. Renesas Technology Corp., All rights reserved. Printed in Japan.

Renesas Technology Corp. Sales Strategic Planning Div. Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan

RENESAS SALES OFFICES

http://www.renesas.com

Refer to "**http://www.renesas.com/en/network**" for the latest and detailed information.

Renesas Technology America, Inc.
450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.
Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7898

Renesas Technology Hong Kong Ltd.
7th Floor, North Tower, World Finance Centre, Harbour City, 1 Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2730-6071

Renesas Technology Taiwan Co., Ltd.
10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

Renesas Technology Singapore Pte. Ltd.
1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.
Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology Malaysia Sdn. Bhd
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jalan Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: <603> 7955-9390, Fax: <603> 7955-9510

SuperH™ Family E10A-USB Emulator Additional Document for User's Manual Supplementary Information on Using the SH7211

Renesas Technology Corp.
2-6-2, Ote-machi, Chiyoda-ku, Tokyo, 100-0004, Japan